EUROPEAN

MIDDLEWARE INITIATIVE

HILA SHELL USER GUIDE

UNICORE Team

Document Version:	1.0.0
EMI Component Version:	v2.2
Date:	April 2011

This work is co-funded by the EC EMI project under the FP7 Collaborative Projects Grant Agreement Nr. INFSO-RI-261611.

Contents

1 Preface **1**

2 Commands **1**

3 General commands **1**

3.1 exit . . . 1
3.2 help . . . 2
3.3 ls . . . 2
3.4 quit . . . 2
3.5 time . . . 2
3.6 version . . . 2

4 Location Commands **3**

4.1 cd . . . 3
4.2 chdir . . . 3
4.3 lcd . . . 3
4.4 lls . . . 3
4.5 lpwd . . . 3
4.6 pwd . . . 3

5 Task Management **4**

5.1 abort . . . 4
5.2 cancel . . . 4
5.3 monitor . . . 4
5.4 status . . . 4

6 Job Management **4**

6.1 submit . . . 4
6.2 start . . . 5
6.3 run . . . 5
6.4 cleanup . . . 5

7 Data handling commands **6**

7.1 Copy Semantics 6
7.2 cat 6
7.3 chmod 6
7.4 copy 7
7.5 cp 7
7.6 import 7
7.7 export 7
7.8 mkdir 7
7.9 move 8
7.10 mv 8
7.11 transfer 8
7.12 rm 8

1 Preface

HiLA Shell is Grid client based on the HiLA API. It was incepted as a demonstrator application to show the features of HiLA in use and to give developers a better understanding of the underlying concepts. At the same time, the commands implemented in the Shell can serve as templates for ones own developments with HiLA.

Nonetheless, HiLA Shell can be used to access Grid resources and act upon them. In the following, the available commands of HiLA Shell will be described.

2 Commands

The first command you should know is `help`. It will list all available commands.

```
:> help
Available commands are:
Available commands are:
cleanup    cat        lls        rm         ls         lcd         ↩
    cancel     run        quit
version    help       time       import     chdir      move        ↩
    copy       export     transfer
monitor    submit     status     mv         lpwd       mkdir       ↩
    exit       pwd        cp
start      chmod      cd         abort
:>
```

Most commands have a `--help` or `-h` option that describes the command and lists possible options and arguments.

We have separated the commands into five categories for documentation purposes.

The Location commands category documents all commands that are related to location like the current location or others. The Task Management category describes all commands that are needed for task management, like aborting or querying the status of a running task. Commands for handling acutal compute jobs are documented in the Job Management category. Data handling commands are described in the Data Handling section. The General category documents all other commands that could not be attributed to a single one of the other categories.

3 General commands

3.1 exit

Exit HiLA Shell.

3.2 help

List available commands.

3.3 ls

List children of the current location.

```
sites :> ls
unicore6:/sites/EMI-UNICOREX_201104071445
sites :>
```

or

```
tasks :> ls
unicore6:/sites/EMI-UNICOREX_201104071445/tasks/de6f68c8-99b4-4a77- ↩
    a873-c94a87e27c96
unicore6:/sites/EMI-UNICOREX_201104071445/tasks/c77f3714-c1c5-4c59- ↩
    befc-7f1191246704
unicore6:/sites/EMI-UNICOREX_201104071445/tasks/2dbd7cb4-4098-429a ↩
    -929f-b092eff82f52
tasks :>
```

3.4 quit

An alias for exit.

3.5 time

Execute a command and print how long it took.

```
tasks :> time run -e /bin/date -s unicore6:/sites/VSGC_201007221554
Fr 23. Jul 13:51:51 CEST 2010

Execution of run took 4269ms.
tasks :>
```

3.6 version

Prints version information about HiLA shell to the console. The output will look similar to this.

```
 :> version
HiLA Shell 2.2  (20110216-0757)
 :>
```

This information is useful if you ask developers for support, e.g. on the UNICORE support mailing list <unicore-support@lists.sourceforge.net>.

4 Location Commands

4.1 cd

Change directory.

HiLA Shell keeps track about the current location. This command resembles command line behavior, where `cd` or `chdir` change the current directory, respectively the current location.

4.2 chdir

An alias for cd.

4.3 lcd

Change the current local diretory. This is useful if you want to use relative local paths for certain commands, e.g. import/export.

4.4 lls

List local directory.

4.5 lpwd

Display current local directory.

```
sites :> lpwd
/home/bjoernh
sites :>
```

4.6 pwd

Display the current (remote) `Location`.

```
EMI-UNICOREX_201104071445 :> pwd
unicore6:/sites/EMI-UNICOREX_201104071445
EMI-UNICOREX_201104071445 :>
```

5 Task Management

5.1 abort

This command aborts a running Task.

```
abort <task location> ...
```

5.1.1 Options

-h, --help Print help message.

5.2 cancel

An alias for abort.

5.3 monitor

Monitor the forthcoming of a task, e.g. import, export, or compute task. Actually, this command works on the `Progressable` interface, thus any `Resource` implementing this interface can be monitored by it.

5.4 status

Display status of `Tasks`.

5.4.1 Options

-l, --log Print the job log.
-h, --help Print help message.

6 Job Management

6.1 submit

Submit a job to a `Site`

6.1.1 Options

-s, --site Site to which to submit the job.
-r, --run Run job right away.
-j, --jsdl JSDL document to be submitted.
-h, --help Print help.

6.2 start

Start a `Job` that is in state NEW.

6.2.1 Options

-b Block on task. Optional argument for blocking limited number of ms.
-h, --help Print this help message.

6.3 run

Run a remote command and display its output.

6.3.1 Options

-arg, --arg Argument
-e, --exec Executable
-s, --site Site to which to submit the job.
-an, --application Application name
-ver, --version Application version
-h, --help Print help message.

6.3.2 Example

The following command will run `/usr/bin/whoami` at the site given by the `-s` option.

```
 :> run -e /usr/bin/whoami -s unicore6:/sites/DEMO- ↩
    SITE_201004221047
demo
 :>
```

6.4 cleanup

Clean up a Job, i.e. remove it from the site. The -a option will export all files from the working directory of the Job to the local export directory, usually ~/.hila2/data/<job id>. The -e option, which can be given multiple times, only exports files that are mentioned by the option. Files can be given with a path relative to the working directory of the job.

6.4.1 Options

-e, --export Export files from working directory. -a, --export-all Export all files. -h, --help Print help message.

7 Data handling commands

7.1 Copy Semantics

Some of the data handling commands deal with the movement of data, i.e. copying or moving files either within a single storage or between two storages. These operations imply that files can already be existing at the target location. Therefore, we describe the semantics of handling these cases here. All copy and data movement commands follow these semantics.

Consider operation

```
op <source file location> <target file location>
```

where `op` is one of the respective commands and `source file location` and `target file location` are valid local or remote file locations.

If target file location does not exist yet, but its parent location does and points to a directory, then the source file is put into that directory and given the name of the source file. If the parent location is an ordinary file, then the operation will fail, as files cannot be put underneath other ordinary files.

If <target file location> points to an already existing file, the following will happen. If <file location2> points to a directory, then the file will be copied into the directory. If <file location2> points to an ordinary file, then that file will only be overwritten, if the -f/--force option is given.

7.2 cat

Output the contents of a remote file to the local console.

```
cat <remote file location>
```

Be careful with `cat`ing binary files, they may render your console unusable.

7.3 chmod

Change file mode bits. File mode bits can only be set for the user, not for the group or other users like you may know from the UNIX command `chmod`.

chmod [-w true/false] [-r true/false] [-x true/false] `<file location>`

7.3.1 Options

-w, --write Set write permission to true/false
-r, --read Set read permission to true/false
-h, --help This help message.
-x, --execute Set executable permission to true/false

7.4 copy

Copy a file within a storage.

copy <file location1> <file location2>

Copy semantics apply.

7.4.1 Options

-f, --force Force overwrite
-r, --recursive recursive operation
-h, --help Print help message.

7.5 cp

An alias for copy.

7.6 import

Import files from local file system to remote storages.

Copy semantics apply.

7.7 export

Export files from remote storages to local file system.

export <file location> <local file path>

Copy semantics apply.

7.8 mkdir

Make directory.

Be sure to give a valid file location. Creating directories does not make sense if you do not create them inside a storage.

7.9 move

Move files within a `Storage`.

7.10 mv

An alias for move.

Copy semantics apply.

7.11 transfer

Transfer a remote file to another file or directory.

transfer <remote:/file/a> <remote:/file/b>

Copy semantics apply.

7.12 rm

Remove files or directories. If you want to remove a directory, the `-r` option needs to be given.

7.12.1 Options

-r, --recursive